BIM Issue Manager

Technical Manual

BIM-IM
ISSUE MANAGER

Ahmed Moenes

Presented by: **Hazem Rashad**

Mostafa El-wahsh

BIM Issue Manager – Technical Manual

1. Executive Technical Overview

The **BIM Issue Manager** is a purpose-built platform that addresses one of the most persistent challenges in modern BIM workflows: the fragmented handling of design and construction issues. Rather than relying on disconnected screenshots, spreadsheets, or generic task tools, this system provides a centralized, model-aware approach to issue tracking — integrated directly with Building Information Models (BIM), enriched with context, and built for team collaboration.

The platform is composed of three main components:

- A **Revit Plugin** that allows users to create issues from within the modeling environment. Each issue captures key metadata such as element IDs, 3D camera positions, and snapshot images of the view, ensuring full traceability.
- A **Standalone Desktop Application** for issue markup and management. This editor includes a built-in 3D IFC viewer, markup tools, team and user management, AI-powered assistance, and a dashboard for statistics and reporting.
- A **Web API and SQL-backed database layer** that ensures synchronization, consistency, and long-term storage. This backend enables multi-project support, user role control, and future expansion into web or mobile clients.

All three layers communicate seamlessly to support an end-to-end issue lifecycle: creation, annotation, assignment, resolution, and tracking. Issues remain linked to the original design context — whether captured from Revit, visualized in 3D, or discussed in the team chat panel.

The system is designed to be both technically solid and user-friendly. It uses a modular architecture based on .NET (WPF for desktop, ASP.NET Core for the API, and Revit API for plugin development), with a normalized SQL schema and clear separation of responsibilities through MVVM and clean architecture principles. The desktop app also includes advanced visualization features using Helix Toolkit and Xbim for IFC, and AI suggestions powered by integrated NLP modules.

With this architecture, the BIM Issue Manager becomes not just a tracker, but a **smart collaboration tool** that embeds coordination deeply into the project lifecycle — improving accountability, reducing resolution times, and helping teams deliver more reliably and transparently.

2. Technology Stack – Built with Purpose

Every component in BIM Issue Manager was carefully chosen — not just for utility, but for cohesion.

Client-Side

Component	Tech	Purpose
Revit Plugin	Revit API (.NET), MVVM	Native creation of issues from BIM context
Desktop App	WPF (.NET 8), MVVM Toolkit, HandyControls	Full-featured offline/online editing + 3D Viewer
3D Viewer	ThatOpenCompany	Load & navigate .ifc models visually
AI Assistant	GPT token	Suggest priorities, autofill fields, summarize context
Charts/Dashboards	LiveCharts2	Track performance, issue density, user stats

Backend

Layer	Stack	Role
API	ASP.NET Core Web API	RESTful backend with full CRUD + filters
Auth	ASP.NET Identity, JWT	Secure role-based login
Data	SQL Server, EF Core	Centralized issue repository
File Handling	Multipart Uploads, Temp Caching	Snapshot/image handling
DI & Validation	Scoped Services, FluentValidation	Reliable business logic enforcement

3. Three-Tier Structure:

1) **Presentation Layer**
    a) Revit Plugin (native context)
    b) WPF Desktop (markup editor, 3D viewer)
    c) Optional Web UI (future expansion)
2) **Application Layer**
    a) API + Services (business logic, orchestration)
    b) DTO mappers, validators, scoped units of work
3) **Data Layer**
    a) SQL Schema with normalized models
    b) Snapshots, issues, metadata, view contexts, users
    c) Safe paths, traceable audit history

4. Key Modules & Innovations

a) Revit Plugin – Contextual Issue Creation

- Creates issues *in situ* from a live BIM view
- Captures camera position, ElementId, UniqueId, ViewId, and snapshot
- Submits via HttpClient to API with full multipart image upload
- Syncs model context to external world with *no data loss*

b) Standalone Desktop App – The Command Center

- Browse and filter issues by project, date, user, status
- Annotate snapshots with drawing tools, stickers, and overlays
- **3D IFC Viewer** — interact with .ifc models in real time
- **AI Assistant** — automatically suggests issue titles, priorities, descriptions
- **Statistics Dashboard** — visual breakdown of issues per user/project/status
- Team & company management — assign issues at any level
- Real-time feedback — flagging, notifications, and status transitions

c) Web API – The Heartbeat

- Complete REST interface for all entities: Issues, Projects, Snapshots, Teams, Labels, Users
- Clean endpoint segmentation: /api/issues, /api/lookup, /api/snapshots/upload
- Entity relationships: 1:N issues → elements, projects → teams
- Upload handlers manage image compression, caching, path generation
- Swagger docs available for integration with external tools

d) Database – Smart, Structured, and Scalable

- Normalized SQL Server schema for clean querying
- Path fields for every snapshot, with Revit + IFC separation
- Priority enums, project IDs, flags, assignments – all structured
- Indexes on priority, creation date, resolution state
- Future-proofed for **multi-tenant isolation** and **cloud scaling**

5. Killer Features That Make It Shine

Feature	Description
Snapshot Pipeline	Instant capture, export, upload, and cloud storage of Revit view
AI Context Assistance	Suggests titles, tags, or actions based on NLP analysis
Flagging + Audit Logs	Track every change, who did what, and why
Teams, Companies, and Users	Full multi-layered assignment model
Discussion Threads	Every issue carries conversations, updates, resolution notes
Dynamic Syncing	Change something in Revit? It shows up in the Desktop. Instantly.

6. Deployment & Hosting

Component	Options
API	Azure App Service & IIS.
SQL Server	Azure SQL, Local VM, or On-prem
Desktop App	Manual deployment
Revit Plugin	Nice3Point

7. Development Standards

- Clean Architecture (Presentation, Application, Domain, Infrastructure)
- SOLID principles
- Generic repository with Unit of Work pattern
- DTO separation
- Full MVVM Toolkit in WPF
- Async/await across all services and data access